INSTALLER APACHE, PHP, IMLIB2, PHP-IMLIB ET MYSQL SOUS LINUX

Jérôme Landré
Institut Universitaire de Technologie,
12, rue de la Fonderie, 71 200 Le Creusot, France

février 2002

1 Introduction

Dans ce petit document, on explique comment installer sous Linux les logiciels suivants :

- Apache : Le serveur web le plus utilisé au monde, gratuit et très efficace.
- PHP : Le langage de scripts côté serveur gratuit et performant grâ ce à toutes ses extensions.
- imlib2 : La librairie graphique imlib pour lire et traiter les fichiers JPEG, PNG...
- php-imlib : La librairie PHP pour utiliser les fonctions imlib2 sous PHP directement dans le code. On ouvre, on manipule et on referme donc les images JPEG et PNG très facilement.
- MySQL : On ne présente plus la base de données relationnelle Mysql qui propose un stockage et une interrogation des données trés efficace même sur de grandes bases (plusieurs millions d'enregistrements). Vous allez me dire qu'on n'a pas besoin de millions d'enregistrement, évidemment, mais qui peut le plus peut le moins... MySQL est gratuit si on ne prend pas le support technique.

Pourquoi rédiger un manuel d'installation alors qu'il existe de nombreuses sources de documentation téléchargeables gratuitement ? Et bien tout simplement parce qu'en toute honnêteté c'est franchement le bordel à installer, à chaque fois il manque une librairie ou la version de celle installée ne convient pas, le fichier tar.gz est mauvais, le rpm refuse de s'installer, bref de nombreuses galères en perspectives pour le commun des mortels (dont je fais partie puisque j'ai essayé pendant un moment toutes les astuces que je vais énoncer par la suite avant de trouver la solution).

Bien sûr, les gens qui connaissent Windows et la facilité d'installer un logiciel ne seront pas interessés par le texte qui suit et diront que Linux est vraiment trop compliqué pour eux. Pourquoi alors s'embêter avec un truc compliqué où on doit toujours compiler du code pour faire fonctionner le système ? Voici quelques points pour justifier ces choix :

- Linux est gratuit : Pas de commentaire spécial sinon que Linux coûte autant en francs qu'en euros : 0 (pas besoin de calculatrice).
- Linux est libre : Linux n'appartient à personne, ce sont des développeurs indépendants de toute société qui font avancer le projet.
- Linux est ouvert : Linux peut tourner sur de nombreux systèmes et lors de la compilation, le code source est le même pour toutes. Tout développement est le bienvenu et les sources sont disponibles.
- Linux marche trop bien : Combien de serveurs web, de serveurs mail, de serveurs tout court fonctionne sous Linux sans planter, sans nécessiter un redémarrage tous les quart d'heures ?
- Linux est multiplateforme : Linux fonctionne sur tous les types d'ordinateurs (PC, Apple Macintosh, stations Sun, stations Silicon Graphics, PocketPC, Psion...) à base de processeurs Intel, AMD, Motorola, Acorn... Peut-on en dire autant de Windows ?
- Je suis cinglé : Je continue à faire du bon vieux code en ligne de commande avec vi (le meilleur éditeur de texte du monde). Mais j'aime bien, pourquoi faire simple quand on peut faire compliqué ?

2 C'est parti !

Dans tout ce qui suit, on considère que l'utilisateur qui effectue les opérations est l'administrateur du système root. En effet, il faut quelques privilèges pour installer un serveur web sur une machine, c'est normal. Donc c'est parti en root...

2.1 Mysql

```
[root@localhost tmp]# tar xvzf mysql-3.23.37.tar.gz
[root@localhost tmp]# cd mysql-3.23.37
[root@localhost mysql-3.23.37]# groupadd mysql
[root@localhost mysql-3.23.37]# useradd -g mysql mysql
[root@localhost mysql-3.23.37]# ./configure --prefix=/home/mysql
[root@localhost mysql-3.23.37]# make
[root@localhost mysql-3.23.37]# make install
[root@localhost mysql-3.23.37]# ./scripts/mysql_install_db
[root@localhost mysql-3.23.37]# chown -R root /home/mysql
[root@localhost mysql-3.23.37]# chown -R mysql /home/mysql/var
[root@localhost mysql-3.23.37]# chgrp -R mysql /home/mysql
--------- Ensuite, deux choix, on lance tout simplement mysqld dans le rc.local au d\'emarra
-- 1 --
[root@localhost mysql-3.23.37]# vi /etc/rc.d/rc.local
#!/bin/sh
#
# This script will be executed *after* all the other init scripts.
```

```
# You can put your own initialization stuff in here if you don't
# want to do the full Sys V style init stuff.

touch /var/lock/subsys/local

/usr/local/mysql/bin/mysql.server start
[root@localhost mysql-3.23.37]#
--------- ou bien on ajoute mysqld comme service de linux
-- 2 --
[root@localhost mysql-3.23.37]# cp /home/mysql/share/mysql/mysql.server /etc/rc.d/init.d
[root@localhost mysql-3.23.37]# cd /etc/rc.d/init.d
[root@localhost mysql-3.23.37]# chkconfig --add mysql.server
--------- Dans les deux cas, il faut configurer mysql pour les utilisateurs et le syst\'eme

/* Il faut ajouter le mot de passe des utilisateurs et des droits sp\'eciaux */

[root@localhost mysql-3.23.37]# vi /etc/ld.so.conf
/usr/kerberos/lib
/usr/X11R6/lib
/usr/lib/qt-2.3.1/lib
/usr/lib/sane
/usr/lib/qt-1.45/lib
/usr/lib/wine
/home/mysql/lib/mysql
[root@localhost mysql-3.23.37]# ldconfig
[root@localhost mysql-3.23.37]#
```

2.2 imlib2

La bibliothèque imlib2 est une réécriture complète de sa précédente version imlib. Elle permet la lecture, la manipulation et l'écriture des images au format JPEG et PNG.

```
[root@localhost tmp]# tar xvzf imlib2-1.0.5.tar.gz
[root@localhost tmp]# cd imlib2-1.0.5
[root@localhost imlib2-1.0.5]# ./configure --prefix=/home/imlib2
[root@localhost imlib2-1.0.5]# make
[root@localhost imlib2-1.0.5]# make install
root@localhost imlib2-1.0.5]# vi /etc/ld.so.conf
/usr/kerberos/lib
/usr/X11R6/lib
/usr/lib/qt-2.3.1/lib
/usr/lib/sane
/usr/lib/qt-1.45/lib
/usr/lib/wine
```

```
/home/mysql/lib/mysql
/home/imlib2/lib
root@localhost imlib2-1.0.5]# ldconfig
--------- Pour tester, il y a un r\'epertoire nomm\'e test dans l'arborescence, il faut test
[root@localhost imlib2-1.0.5]# cd test
[root@localhost test]# make
[root@localhost test]# ./imlib2
--------- Si toutefois, ca ne voulait pas marcher comme sur ma version, il faut bricoler un
[root@localhost test]# cp -R /home/imlib2/lib/* /usr/lib/
[root@localhost test]# make
[root@localhost test]# ./imlib2
[root@localhost test]# cd ..
[root@localhost imlib2-1.0.5]#
```

2.3 apache

On installe apache en tant que module du système, ce qui facilite les choses et ne surcharge pas le serveur inutilement.

```
[root@localhost tmp]# tar xvzf apache_1.3.23.tar.gz
[root@localhost tmp]# cd apache_1.3.23/
[root@localhost apache_1.3.23]# ./configure --prefix=/home/www --enable-module=so
[root@localhost apache_1.3.23]# make
[root@localhost apache_1.3.23]# make install
[root@localhost apache_1.3.23]# vi /home/www/conf/httpd.conf
...
#
# DirectoryIndex: Name of the file or files to use as a pre-written HTML
# directory index.  Separate multiple entries with spaces.
#
<IfModule mod_dir.c>
    DirectoryIndex index.php index.html
</IfModule>
...
    #
    # AddType allows you to tweak mime.types without actually editing it, or to
    # make certain files to be certain types.
    #
    AddType application/x-tar .tgz
    AddType application/x-httpd-php .php
...
[root@localhost apache_1.3.23]# vi /etc/rc.d/rc.local
#!/bin/sh
#
```

```
# This script will be executed *after* all the other init scripts.
# You can put your own initialization stuff in here if you don't
# want to do the full Sys V style init stuff.

touch /var/lock/subsys/local

echo "demarrage de Mysql..."
/usr/local/mysql/bin/mysql.server start
echo "Demarrage d'Apache..."
/usr/local/apache/bin/apachectl start
echo
[root@localhost apache_1.3.23]#
```

2.4 PHP et php-imlib

Alors là, ça ne rigole plus du tout, c'est la partie la plus difficile!!! Il faut suivre avec atteznton et ne pas rater un épisode....

```
[root@localhost tmp]# tar xvzf php-4.1.1.tar.gz
[root@localhost tmp]# tar xvzf php_imlib-0.3.tar.gz
[root@localhost tmp]# cd php-4.1.1
[root@localhost php-4.1.1]# cd ext
[root@localhost ext]# ./ext_skel --extname=imlib
Creating directory imlib
Creating basic files: config.m4 Makefile.in .cvsignore imlib.c php_imlib.h CREDITS EXPERIMEN

To use your new extension, you will have to execute the following steps:

1.  $ cd ..
2.  $ vi ext/imlib/config.m4
3.  $ ./buildconf
4.  $ ./configure --[with|enable]-imlib
5.  $ make
6.  $ ./php -f ext/imlib/imlib.php
7.  $ vi ext/imlib/imlib.c
8.  $ make

Repeat steps 3-6 until you are satisfied with ext/imlib/config.m4 and
step 6 confirms that your module is compiled into PHP. Then, start writing
code and repeat the last two steps as often as necessary.

[root@localhost ext]# cp -rf ../../imlib/* imlib/
[root@localhost ext]# cd ../..
[root@localhost tmp]# chown -R root.root php-4.1.1
[root@localhost ext]# cd ../..
```

```
[root@localhost tmp]# chown -R root.root php-4.1.1
[root@localhost tmp]# cd php-4.1.1
[root@localhost php-4.1.1]# ./buildconf
buildconf: checking installation...
buildconf: autoconf version 2.13 (ok)
buildconf: automake version 1.4-p5 (ok)
buildconf: libtool version 1.4 (ok)
rebuilding Makefile templates
rebuilding configure
rebuilding main/php_config.h.in
[root@localhost php-4.1.1]# ./configure --with-imlib=/home/imlib2 --with-gd --with-gmp --wit
[root@localhost php-4.1.1]# make
make[3]: Entre dans le rpertoire `/tmp/php-4.1.1/ext/imlib'
/bin/sh /tmp/php-4.1.1/libtool --silent --mode=compile gcc  -I. -I/tmp/php-4.1.1/ext/imlib -
imlib.c:104: warning: initialization makes integer from pointer without a cast
imlib.c:104: initializer element is not computable at load time
imlib.c:104: (near initialization for `imlib_module_entry.size')
imlib.c:105: warning: initialization makes integer from pointer without a cast
imlib.c:106: warning: initialization makes integer from pointer without a cast
imlib.c:106: initializer element is not computable at load time
imlib.c:106: (near initialization for `imlib_module_entry.zend_debug')
imlib.c:107: warning: initialization makes integer from pointer without a cast
imlib.c:108: warning: initialization from incompatible pointer type
imlib.c:109: warning: initialization from incompatible pointer type
imlib.c:110: warning: initialization from incompatible pointer type
make[3]: *** [imlib.lo] Erreur 1
make[3]: Quitte le rpertoire `/tmp/php-4.1.1/ext/imlib'
make[2]: *** [all-recursive] Erreur 1
make[2]: Quitte le rpertoire `/tmp/php-4.1.1/ext/imlib'
make[1]: *** [all-recursive] Erreur 1
make[1]: Quitte le rpertoire `/tmp/php-4.1.1/ext'
make: *** [all-recursive] Erreur 1
[root@localhost php-4.1.1]# cd ext/imlib
[root@localhost imlib]# vi imlib.c
--- AVANT ---
zend_module_entry imlib_module_entry = {
        "imlib",
        imlib_functions,
        PHP_MINIT(imlib),
        NULL,
        PHP_RINIT(imlib),
        PHP_RSHUTDOWN(imlib),
        PHP_MINFO(imlib),
        STANDARD_MODULE_PROPERTIES
};
--- APRES ---
```

```
static unsigned char IMLIB_VERSION[] = { "php_imlib_0.3"} ;

zend_module_entry imlib_module_entry = {
#if ZEND_MODULE_API_NO >= 20010901
        STANDARD_MODULE_HEADER,
#endif
        "imlib",
        imlib_functions,
        PHP_MINIT(imlib),
        NULL,
        PHP_RINIT(imlib),
        PHP_RSHUTDOWN(imlib),
        PHP_MINFO(imlib),
#if ZEND_MODULE_API_NO >= 20010901
        IMLIB_VERSION,
#endif
        STANDARD_MODULE_PROPERTIES
};
[root@localhost imlib]# cd ../..
[root@localhost php-4.1.1]# make
[root@localhost php-4.1.1]# make install
```

2.5 Conclusion

On peut tout de même dire qu'on est content quand ça marche à la fin. Enfin j'espère que ce petit manuel vous sera utile pour le résolution d'éventuels problèmes liés à l'installation d'un des logiciels ci-dessus (ou de tous à la fois !). Alors bon courage, et ne perdez pas espoir, vous n'êtes jamais seul dans la longue liste des utilisateurs de logiciels libres, les ressources sont disponibles pour vous venir en aide et souvenez-vous que la communauté du logiciel libre compte sur vous si vous avez un peu de temps à lui consacrer...